Collaborating on Efficient Database Models

Part 1: Office Roleplay Dialogue

Scenario: A Database Administrator, Naomi, is working with a Software Developer, Daniel, to design an efficient **database model** for a new application.

Daniel: Hey Naomi, we need to finalize the **database model** for the new customer management system. Do you have time to review the structure with me?

Naomi: Of course! I already created an **ERD (Entity Relationship Diagram)** to visualize how the tables are connected.

Daniel: That's great! I want to make sure we define the **relationships** correctly between customers, orders, and payments.

Naomi: Absolutely. We should establish one-to-many **relationships** between customers and orders, and between orders and payments.

Daniel: That makes sense. What about data redundancy? Should we apply **normalization** to minimize duplicate data?

Naomi: Yes, but we should balance **normalization** with performance. Too many joins can slow down queries, so we need to structure it carefully.

Daniel: Agreed. Also, after we launch, we may need to adjust the structure. How do we handle that efficiently?

Naomi: We can use **refactoring** to optimize the database over time without disrupting existing functionality.

Daniel: Good point. So, if we need to merge tables or modify columns, we do it through **refactoring** while ensuring minimal downtime?

Naomi: Exactly! We'll also monitor performance to see if any adjustments are needed later.

Daniel: Sounds like a solid plan. Let's finalize the **ERD**, and I'll update the documentation accordingly.

Naomi: Perfect. Once that's done, we can start implementing the database structure.

Part 2: Comprehension Questions

1. What tool did Naomi use to visualize the database structure?
(A) A spreadsheet
(B) A project management software
(C) A text document
(D) ERD (Entity Relationship Diagram)

2. Why is normalization important in database design?
(A) It speeds up network connections
(B) It reduces data redundancy
(C) It removes all foreign keys
(D) It replaces tables with spreadsheets

3. What does Daniel want to ensure about relationships in the database?
(A) That they are correctly defined between customers, orders, and payments

(B) That they are removed to simplify the structure
(C) That they all use the same primary key
(D) That they exist only in temporary tables

4. What is refactoring used for in database management?
(A) To delete all unnecessary records
(B) To increase storage capacity
(C) To optimize the database without disrupting existing functionality
(D) To merge all tables into one large table

Part 3: Key Vocabulary Definitions in Japanese

1. **Database Model (データベースモデル)** – データの構造や関係を定義し、どのようにデータを保存・管理するかを決める設計。
2. **ERD (Entity Relationship Diagram) (ERD・エンティティリレーションシップ図)** – データベースのテーブルとその関係を視覚的に表す図。
3. **Relationships (リレーションシップ・関係性)** – データベース内のテーブル同士のつながりを示す概念。
4. **Normalization (正規化)** – データの重複を減らし、整合性を保つためにデータベースを最適化する手法。

5. **Refactoring (リファクタリング)** – パフォーマンスを向上させるために、データベースの構造を整理・最適化すること。

Part 4: Questions & Correct Answers

1. **What tool did Naomi use to visualize the database structure?**
   ✅ (D) ERD (Entity Relationship Diagram)
2. **Why is normalization important in database design?**
   ✅ (B) It reduces data redundancy
3. **What does Daniel want to ensure about relationships in the database?**
   ✅ (A) That they are correctly defined between customers, orders, and payments
4. **What is refactoring used for in database management?**
   ✅ (C) To optimize the database without disrupting existing functionality